



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У
НОВОМ САДУ



Даниел Кнежевић

**Реализација GWT клијент апликације
за приступ послужиоцу за
конфигурисање STB коришћењем MVP
пројектног узорка**

ДИПЛОМСКИ РАД
- Основне академске студије -

Нови Сад, 2013



КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:		
Идентификациони број, ИБР:		
Тип документације, ТД:	Монографска документација	
Тип записа, ТЗ:	Текстуални штампани материјал	
Врста рада, ВР:	Завршни (Bachelor) рад	
Аутор, АУ:	Даниел Кнежевић	
Ментор, МН:	др Иштван Пап	
Наслов рада, НР:	Реализација GWT клијент апликације за приступ послужиоцу за конфигурисање STB коришћењем MVP пројектног узорка	
Језик публикације, ЈП:	Српски / латиница	
Језик извода, ЈИ:	Српски	
Земља публиковања, ЗП:	Република Србија	
Уже географско подручје, УГП:	Војводина	
Година, ГО:	2013	
Издавач, ИЗ:	Ауторски репринт	
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6	
Физички опис рада, ФО: (поплавља/страна/цитата/табела/слика/графика/прилога)	7/32/0/7/22/0/0	
Научна област, НО:	Електротехника и рачунарство	
Научна дисциплина, НД:	Рачунарска техника	
Предметна одредница/Кључне речи, ПО:	GWT, MVP, TR069, веб апликација	
УДК		
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад	
Важна напомена, ВН:		
Извод, ИЗ:	У овом раду је реализован GWT клијент апликација за приступ послужиоцу за конфигурисање STB коришћењем MVP пројектног узорка.	
Датум прихватања теме, ДП:		
Датум одбране, ДО:		
Чланови комисије, КО:	Председник:	др Никола Теслић
	Члан:	др Милан Бјелица
	Члан, ментор:	др Иштван Пап
		Потпис ментора



KEY WORDS DOCUMENTATION

Accession number, ANO:		
Identification number, INO:		
Document type, DT:	Monographic publication	
Type of record, TR:	Textual printed material	
Contents code, CC:	Bachelor Thesis	
Author, AU:	Daniel Knežević	
Mentor, MN:	Ištván Pap, PhD	
Title, TI:	Implementation of the GWT web front-end application of the STB configuration server by means of MVP design pattern.	
Language of text, LT:	Serbian	
Language of abstract, LA:	Serbian	
Country of publication, CP:	Republic of Serbia	
Locality of publication, LP:	Vojvodina	
Publication year, PY:	2013	
Publisher, PB:	Author's reprint	
Publication place, PP:	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, PD: (chapters/pages/ref./tables/pictures/graphs/appendices)	7/32/0/7/22/0/0	
Scientific field, SF:	Electrical Engineering	
Scientific discipline, SD:	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, S/KW:	GWT, MVP, TR069, web application	
UC		
Holding data, HD:	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, N:		
Abstract, AB:	In this paper the GWT web front-end of the STB configuration is implemented. The solution is designed for configuring consumer electronics devices using MVP design pattern.	
Accepted by the Scientific Board on, ASB:		
Defended on, DE:		
Defended Board, DB:	President:	Nikola Teslić, PhD
	Member:	Milan Bjelica, PhD
	Member, Mentor:	Ištván Pap, PhD
		Menthor's sign

SADRŽAJ

1.	Uvod.....	1
2.	Teorijske osnove	3
2.1	TR-069.....	3
2.2	Google Web Toolkit (GWT)	4
2.3	Model-View-Presenter (MVP) projektni uzorak	5
3.	Koncept rešenja.....	7
3.1	Klijentski deo aplikacije	8
3.1.1	Jezgro	8
3.1.2	View	9
3.1.3	Presenter.....	10
3.1.4	EventBus mehanizam i događaji.....	10
3.2	Deo aplikacije za komunikaciju sa poslužiocem.....	10
4.	Programsko rešenje	12
4.1	Moduli i metode GWT aplikacije	12
4.1.1	Paket com.rtrk.insight.acs.gwt	13
4.1.2	Paket com.rtrk.insight.acs.gwt.client	13
4.1.3	Paket com.rtrk.insight.acs.gwt.client.async	13
4.1.4	Paket com.rtrk.insight.acs.gwt.client.engine	13
4.1.4.1	AppController	13
4.1.4.2	AppEventBus	14
4.1.4.3	IbasicDisplay	15
4.1.4.4	PlaceController	15
4.1.4.5	Presenter	15

4.1.4.6	StatusManager	16
4.1.4.7	View.....	16
4.1.5	Paket com.rtrk.insight.acs.gwt.client.engine.security	16
4.1.6	Paket com.rtrk.insight.acs.gwt.client.event	16
4.1.7	Paket com.rtrk.insight.acs.gwt.client.presenter	18
4.1.8	Paket com.rtrk.insight.acs.gwt.client.view	19
4.1.8.1	CpeInfoView.....	20
4.1.8.2	CpeListView	20
4.1.8.3	DeviceListView	21
4.1.8.4	EditCpeView.....	22
4.1.8.5	FirmwareListView	22
4.1.8.6	FirmwareUp grade View	22
4.1.8.7	MainView	23
4.1.8.8	MenuView	24
4.1.8.9	ModifyDevice View	24
4.1.8.10	NotificationView	25
4.1.8.11	OwnerInfoView	25
4.1.8.12	TopView	25
4.1.8.13	UsersView.....	25
4.1.9	Paket com.rtrk.insight.acs.gwt.servlets.rpc	26
4.1.10	Paket com.rtrk.insight.acs.gwt.shared	26
5.	Rezultati	28
6.	Zaključak.....	31
7.	Literatura	32

SPISAK SLIKA

Slika 2.1 Protokol stek	3
Slika 2.2 Grafički prikaz TR-069 protokola.....	4
Slika 3.1 Primer rešenja grafičke korisničke sprege komercijanog ACS	7
Slika 3.2 Arhitektura klijentskog dela aplikacije	8
Slika 3.3 Osnovne komponente MVP principa i veze između njih	9
Slika 3.4 Dijagram realizacije RPC.....	11
Slika 3.5 Dijagram redosleda komunikacije sa poslužiocem.....	11
Slika 4.1 CpeInfoView	20
Slika 4.2 CpeListView	21
Slika 4.3 DeviceListView	21
Slika 4.4 EditCpeView	22
Slika 4.5 FirmwareListView	22
Slika 4.6 FirmwareUpgradeView.....	23
Slika 4.7 MainView.....	23
Slika 4.8 MenuView.....	24
Slika 4.9 ModifyDeviceView.....	24
Slika 4.10 OwnerInfoView	25
Slika 4.11 TopView.....	25
Slika 4.12 UsersView	26
Slika 5.1 Selenium IDE	28
Slika 5.2 Rezultati ispitivanja.....	29
Slika 5.3 Vreme odziva na komande iz aplikacije	30

SPISAK TABELA

Tabela 4.1 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.engine	13
Tabela 4.2 Spisak sprega paketa com.rtrk.insight.acs.gwt.client.engine.security.....	16
Tabela 4.3 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.event	17
Tabela 4.4 spisak sprega paketa com.rtrk.insight.acs.gwt.client.event	18
Tabela 4.5 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.presenter	19
Tabela 4.6 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.view	19
Tabela 4.7 Spisak klasa paketa com.rtrk.insight.acs.gwt.shared	27

SKRAĆENICE

GWT	- <i>Google Web Toolkit</i>
MVP	- <i>Model View Presenter</i>
ACS	- <i>Auto-Configuration Server</i> , poslužilac za automatsku konfiguraciju
CPE	- <i>Customer-Premises Equipment</i> , potrošačka oprema
TR-069	- <i>Technical Report 069</i> , tehnički izveštaj 069
JRE	- <i>Java Runtime Environment</i>
UI	- <i>User Interface</i> , korisnička sprega
RPC	- <i>Remote Procedure Call</i> , poziv udaljene metode
BER	- <i>Bit Error Rate</i> , procenat bita primljenih sa greškom u odnosu na ukupan broj primljenih bitova

HTTP	- <i>Hypertext Transfer Protocol</i>
AJAX	- <i>Asynchronous JavaScript and XML</i>
HTML	- <i>HyperText Markup Language</i>
XML	- <i>Extensible Markup Language</i>
CSS	- <i>Cascading Style Sheets</i>

1. Uvod

Potrošačka oprema u današnje vreme postaje sve složenija i masovnija. Nudi se sve više usluga zahvaljujući širokopojasnom (eng. broadband) pristupu. Kompleksnost dostave i upravljanja širokopojasnih usluga povećava troškove operatera. Kao rešenje pojavio se protokol TR-069, koji u potpunosti otklanja teret konfiguracije potrošačke opreme, jer se sve usluge obavljaju uz pomoć konfiguracionih poslužioca. Konfiguracioni poslužioci predstavljaju gradivni element protokola za daljinsko upravljanje uređajima. Oni omogućuju izvršavanje funkcija na krajnjem uređaju sa ili bez znanja njegovog vlasnika.

U radu je opisano rešenje realizacije web klijent aplikacije za konfigurisanje potrošačke opreme (CPE) posredstvom poslužioca za automatsku konfiguraciju uređaja (ACS). Korišćen je Google Web Toolkit (GWT) skup alata u Java programskom jeziku i Model-View-Presenter (MVP) projektni uzorak radi lakšeg razvoja aplikacije. Rešenje je realizovano korišćem JBoss AS 7 i Insight ACS 1.0 API-a. Obezbeđen je prikaz spiska CPE uređaja sa mogućnošću pretraživanja po vrednosti parametara ili identifikatoru, te podrška za prikaz i ažuriranje programske podrške na odabranoj grupi CPE, kao i mogućnost za izmenu pojedinačnih CPE parametara.

Ispravnost rada aplikacije je verifikovana izvršavanjem skupa odgovarajućih ispitnih slučajeva. Ispitivanje je obavljeno korišćenjem Insight ACS 1.0 i set-top boks uređaja sa podrškom za TR-069 protokol.

Rad se sastoji od sledećih celina:

1. Teorijske osnove - pokrivaju osnove GWT klijent aplikacije.
2. Koncept rešenja - objašnjavaju veze među modulima i njihovu namenu.
3. Programsko rešenje - opis svih metoda u rešenju sa ulaznim parametrima i povratnim vrednostima.
4. Rezultati - prikazani su rezultati testiranja.

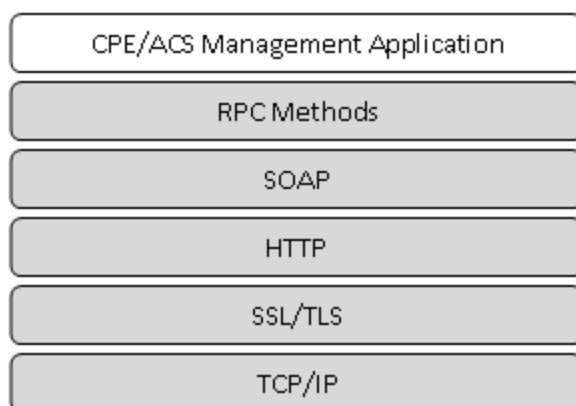
5. Zaključak - pokriva ispunjenost zadatka.

2. Teorijske osnove

U ovom poglavlju izložene su osnovne informacije o TR-069 protokolu i korišćenim tehnologijama kao što su Google Web Toolkit (GWT), skup alata i Model-View-Presenter (MVP), projektni uzorak.

2.1 TR-069

TR-069 je tehnička specifikacija, poznata i kao CWMP (eng. CPE Wan Management Protocol) izdata od strane Broadband Forum. TR-069 opisuje protokol aplikacionog nivoa koji omogućuje daljinsko upravljanje krajnjim korisničkim uređajima potrošačke elektronike slika 2.2. Ovaj protokol omogućuje dvosmernu komunikaciju između entiteta. Komunikacija je bazirana na SOAP/HTTP protokolu. U standardu su propisane dve grupe uređaja: CPE (eng. Customer Premises Equipment) i ACS (eng. Auto-Configuration Server). CPE (eng. Customer Premises Equipment) su krajnji uređaji, njima se upravlja putem centrale ACS (eng. Auto-Configuration Server). Arhitektura protokola je slojevita, svaki sloj predstavlja jedan od standardnih protokola za razmenu poruka između tačaka. Arhitektura protokola je prikazana na slici 2.1.



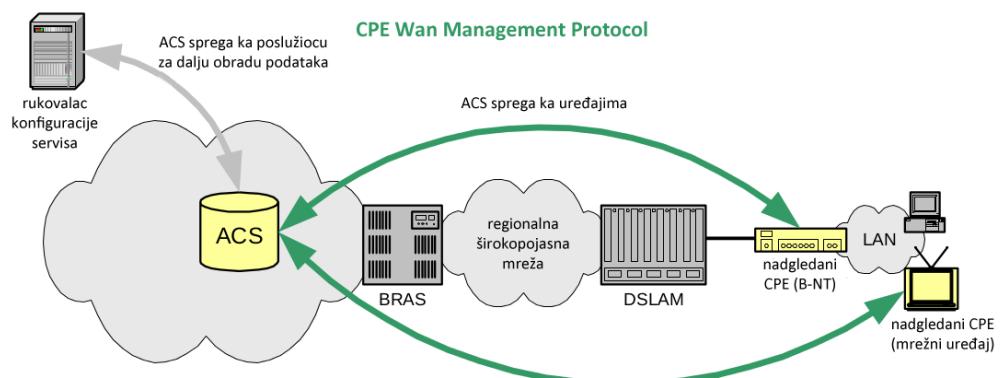
Slika 2.1 Protokol stek

Komunikacija između poslužioca i krajnjih uređaja se odvija u sesijama. Sadržaj poruka, njihova sintaksa i semantika su prethodno definisani. U komunikaciji postoje poruke koje su obavezne i one koje su neobavezne.

Komunikacija je zaštićena SSL enkripcijom koja je sastavni deo HTTPS komunikacionog protokola. Moguća je komunikacija bez SSL enkripcije ali kod sistema sa velikim brojem korisnika to nije preporučljivo.

Preko koncepta modela podataka moguće je prilagoditi komunikaciju i upravljanje parametrima mreže i krajnjih uređaja. Upravo ovaj koncept je doprineo definisanju standardizovanih modela od kojih je jedan specifičan za set-top-boks uređaje (TR-135). Ovakva koncepcija standarda je dovela do toga da nadležne institucije za razvijanje standarda u oblastima digitalne televizije i mrežnih tehnologija predlože TR-069 kao primarno rešenje za upravljanje i nadgledanje krajnjih uređaja.

Osnovne operacije TR-069 protokola su inicijalna konfiguracija uređaja, praćenje statusa uređaja, postavljanje nove programske podrške na krajnjim uređajima, praćenje vrednosti parametara, postavljanje vrednosti nekih parametara, dijagnosika itd.



Slika 2.2 Grafički prikaz TR-069 protokola

2.2 Google Web Toolkit (GWT)

Google Web Toolkit (GWT) je skup alata, otvorenog koda koji omogućava razvijanje i održavanje složenih JavaScript front-end aplikacija u Java programskom jeziku. Osim nekoliko nativnih biblioteka, sve je Java izvorni kod koji se može prevoditi na bilo kojoj podržanoj platformi sa uključenim GWT Ant datotekama za prevođenje. GWT je licenciran pod Apache licencom verzije 2.0.

GWT omogućava ponovnu upotrebu koda (eng. code reuse), jednostavnu realizaciju poziva udaljenih metoda, rukovanje istorijom, apstrakciju korisničke sprege, internacionalizaciju i prenosivost aplikacija na različite pretraživače.

Korišćenjem GWT alata programeri mogu razvijati i otklanjati greške u Ajax aplikacijama u Java programskom jeziku korišćenjem Java razvojnih alata po svom izboru. Kada je aplikacija razvijena GWT prevodilac prevodi Java aplikaciju u samostalne JavaScript datoteke koje su optimizovane i prilagođene ciljnom pregledaču web stranica.

Glavne komponente su:

- GWT Java u JavaScript prevodilac – omogućava prevođenje izvornog koda pisanoog u Java programskom jeziku u JavaScript programski jezik.
- GWT razvojni režim – omogućava programerima da izvršavaju GWT aplikacije u razvojnog režimu, to jest aplikacija se pokreće kao Java aplikacija u Java virtuelnoj mašini bez prevođenja u JavaScript.
- Biblioteka za emulaciju JRE – JavaScript implementacije često korišćenih klasa iz standardne Java biblioteke (kao što je većina java.lang paketa klasa i deo java.util paketa klasa).
- GWT Web UI biblioteka – sadrži skup posebnih sprega i klasa za kreiranje komponenti grafičke korisničke sprege.

Karakteristike:

- Dinamične i ponovo upotrebljive komponente korisničke sprege
- Jednostavan RPC mehanizam
- Upravljanje istorijom pretraživača
- Integracija JUnit-a
- Podrška za otklanjanje grešaka u potpunosti u Java programskom jeziku
- Podrška za internacionalizaciju i lokalizaciju
- Programeri mogu kreirati svoje aplikacije kao objektno-orijentisane pošto koriste Java programski jezik.

2.3 Model-View-Presenter (MVP) projektni uzorak

Pri kreiranju složenih aplikacija postoje prepreke, koje ne zaobilaze ni GWT. Dok više programera radi istovremeno na istom delu izvornog koda postoji velika verovatnoća da će kod biti loše organizovan. Da bi se ovi problemi rešili uvode se projektni uzorci koji stvaraju odvojene oblasti odgovornosti u projektu.

Postoje različiti projektni uzorci kao što su Presentation-abstraction-control, Model-view-controller, Model-view-presenter itd. Dok svaki projektni uzorak ima svoje prednosti Google je ustanovio da MVP arhitektura najbolje funkcioniše kada se razvijaju GWT aplikacije iz dva razloga. Najpre MVP projektni uzorak, poput drugih projektnih uzoraka, razdvaja razvoj tako da više programera mogu da rade istovremeno. Drugo, ovaj projektni uzorak omogućuje minimalan

skup GWT ispitnih slučajeva koji se oslanjaju na prisustvo pretraživača i da za veći deo izvornog koda pišu lagane i brze JRE ispitne slučajeve koji ne zahtevaju pretraživač.

Najvažniji deo uzorka je razdvajanje funkcionalnosti u komponente koje čine logički smisao, ali u slučaju GWT-a pažnja je usmerena na to da se View (pregled) učini što jednostavnijim kako bi se smanjilo oslanjanje na skup GWT ispitnih slučajeva i smanjilo vreme provedeno na ispitivanju.

Kreiranje MVP bazirane aplikacije može biti jednostavno ako se shvate osnove projektnog uzorka.

Osnovne komponente MVP projektnog uzorka su:

- Model
- View
- Presenter

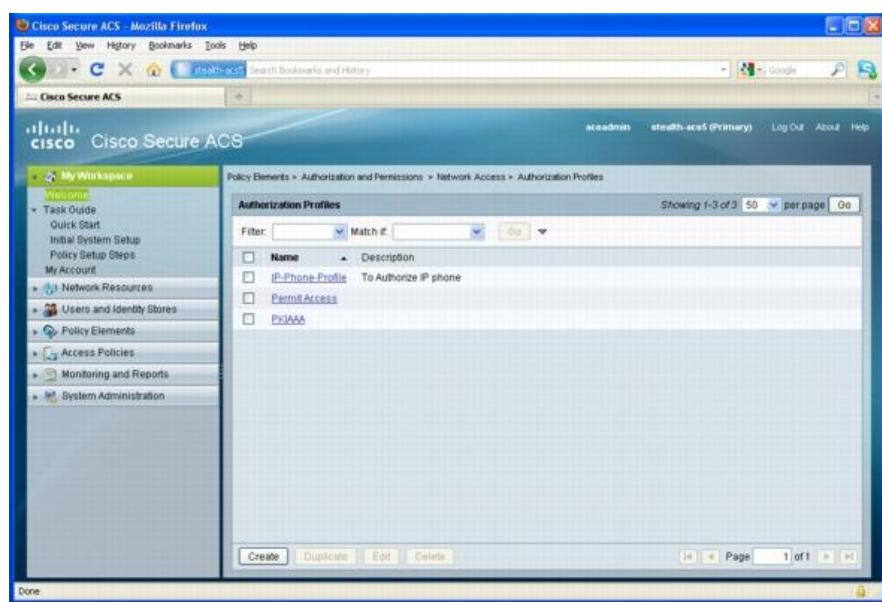
Model obuhvata poslovne objekte.

View sadrži komponente grafičke korisničke sprege koje čine aplikaciju. Sadrži tabele, labele, dugmad itd. Pregledi su odgovorni za izgled komponenti korisničke sprege i nemaju pojam modela. To znači da View ne zna šta treba prikazati, već zna da ima na primer dve tabele, jedno polje za unos i dva dugmeta.

Presenter sadrži logiku aplikacije, uključuje rukovanje istorijom i sinhronizaciju sa poslužiocem putem RPC. Po pravilu za svaki View postoji odgovarajući Presenter koji upravlja prikazom i rukovanje događajima koji potiču od komponenti korisničke sprege.

3. Koncept rešenja

Prilikom projektovanja grafičke korisničke sprege za pristup poslužiocu za konfiguraciju STB treba imati u vidu da je ona namenjena operaterima koji nadgledaju mrežu uređaja. Aplikacija treba da bude jednostavna, da komande budu lako uočljive i podaci jasno prikazani. Uobičajen izgled grafičke korisničke sprege se sastoji iz tri dela: u gornjem delu se nalaze zaglavje i komande za odjavljivanje trenutno prijavljenog korisnika, sa leve strane se nalazi meni za kretanje kroz aplikaciju, a u središnjem delu vidi se sadržaj izabrane stranice. Na slici 3.1 se može videti primer najčešće grafičke korisničke sprege, što je usvojeno i u okviru realizovanog rešenja.



Slika 3.1 Primer rešenja grafičke korisničke sprege komercijanog ACS

Arhitektura projekta se može podeliti u dve osnovne celine:

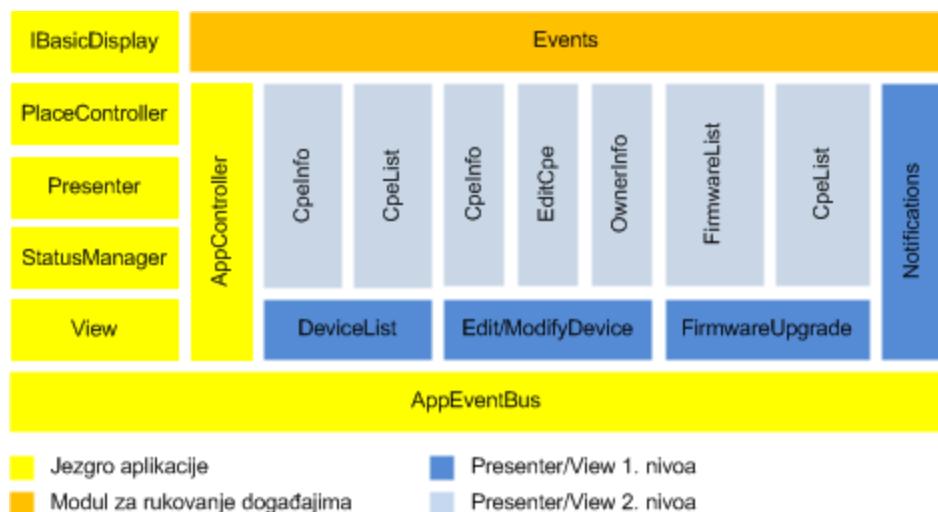
- Klijentski deo aplikacije
- Deo aplikacije za komunikaciju sa poslužiocem

3.1 Klijentski deo aplikacije

Klijentski deo aplikacije sadrži module za realizaciju grafičke korisničke sprege slika 3.2. Arhitektura aplikacije se može podeliti u četiri grupe:

- Jezgro aplikacije – Čini osnovne module aplikacije za realizaciju po MVP principima.
- Modul za rukovanje događajima – Obezbeđuje objavljivanje događaja aplikacije i korisničkih akcija.
- Presenter/View prvog nivoa – Predstavljaju koristan sadržaj stranice ili obezbeđuju kontejner za ugnježdavanje više stranica u jednu.
- Presenter/View drugog nivoa – Predstavljaju najmanju jedinicu stranice koja se prikazuje u okviru Presenter/View prvog nivoa.

U okviru programskog rešenja svaki modul će biti detaljnije opisan.



Slika 3.2 Arhitektura klijentskog dela aplikacije

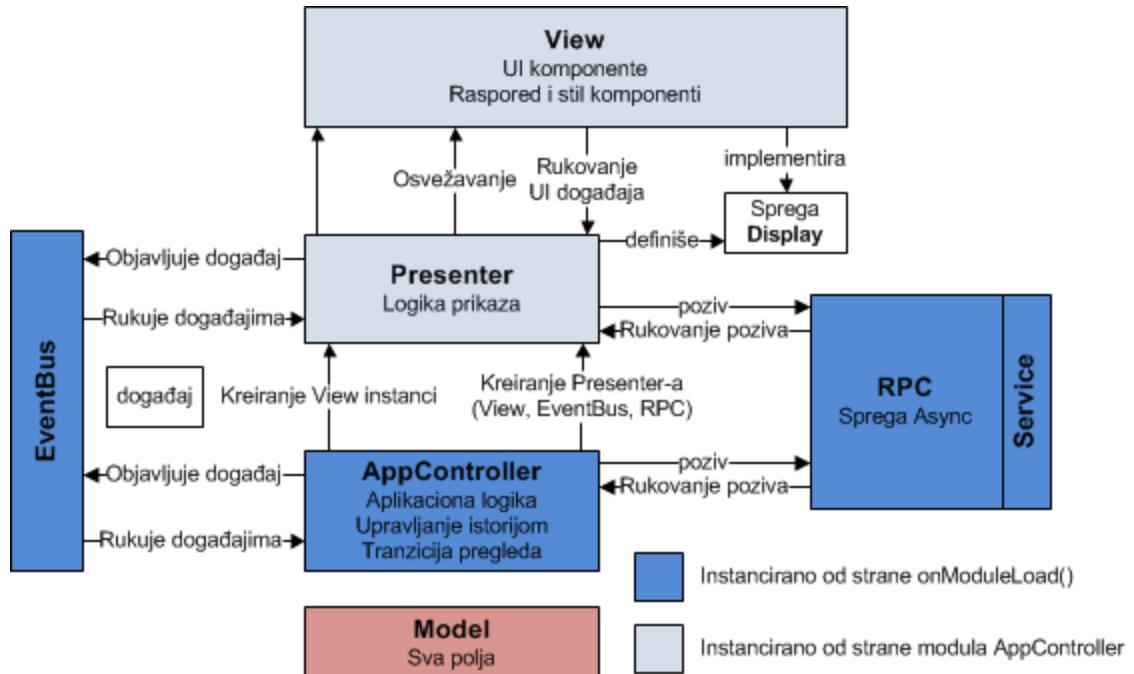
3.1.1 Jezgro

Moduli MVP jezgra su:

- AppController – Predstavlja komponentu u kojoj se inicijalizuju svi moduli View i Presenter.
- EventBus – Namena modula je obezbeđivanje jedinstvenog mehanizma komunikacije između modula Presenter.
- Display – Zadatak ovog modula je povezivanje modula View i Presenter. Povezivanje modula View i Presenter se sastoji iz dva koraka. Presenter definise Display spregu sa svim potrebnim metodama, potom View implementira definisane metode.

- Presenter – Zadatak ovog modula je komunikacija sa poslužiocem putem RPC i sadrži logiku određenog modula View.
- View – Zadatak ovog modula je realizacija grafičkog dela aplikacije, definiše se raspored komponenti grafičke korisničke sprege.

Osnovne komponente MVP principa i veze između njih su prikazane na slici 3.3.



Slika 3.3 Osnovne komponente MVP principa i veze između njih

3.1.2 View

Modul View predstavlja deo aplikacije koji je vidljiv na ekranu. Komponente modula View se kreiraju pomoću GWT alata UiBinder. UiBinder omogućuje mehanizam povezivanja deklarativnog opisa korisničke sprege sa kodom pisanim u Java jeziku. Prednost korišćenja alata UiBinder je jednostavnost projektovanja i izmene korisničke sprege.

Kreiranje komponenti modula View se sastoji iz sledećih koraka:

1. Kreiranje XML datoteke koja sadrži deklarativan opis korisničke sprege. Ime datoteke treba da se završava sa .ui.xml. U ovoj datoteci se uz pomoć HTML, CSS i XML jezika opisuje raspored i izgled komponenti korisničke sprege.
2. Kreiranje Java datoteke koja služi za povezivanje sa XML datotekom.
3. Povezivanje Java i XML komponenti pomoću UiBinder sprege. Posle ovog koraka obezbeđen je pristup komponentama iz Java koda.

3.1.3 Presenter

Modul Presenter predstavlja srednji sloj MVP projektnog uzorka. Njegov zadatak je povezivanje modela podataka sa grafikom vidljivom na ekranu. U okviru ovog modula vrši se komunikacija sa poslužiocem putem RPC sprege, obrađuju se događaji i upravlja se logikom UI komponenti.

3.1.4 EventBus mehanizam i događaji

Pri kreiranju složenih aplikacija jedan od osnovnih ciljeva je modularnost, to jest zaokruživanje funkcionalnosti jednog dela aplikacije u celinu tako da drugi delovi aplikacije ne znaju kako radi. Modularnost se često narušava kada moduli trebaju međusobno komunicirati. Jedan od rešenja ovog problema je korišćenje GWT *event bus* mehanizma koji omogućava komunikaciju između modula. EventBus predstavlja magistralu kroz koju se šalju događaji (eng. events). Komunikacija se zasniva na tome da modul koji objavi događaj ne zna kome je namenjen, već se zainteresovani moduli trebaju pretplatiti na do gađaje.

Događaj se sastoji iz dva dela:

1. Klasa koja sadrži opis događaja. Pri objavljuvanju događaja objekat klase će se proslediti na magistrali događaja.
2. Sprega koja sadrži metodu koju implementira modul koji je preplaćen na dati događaj.

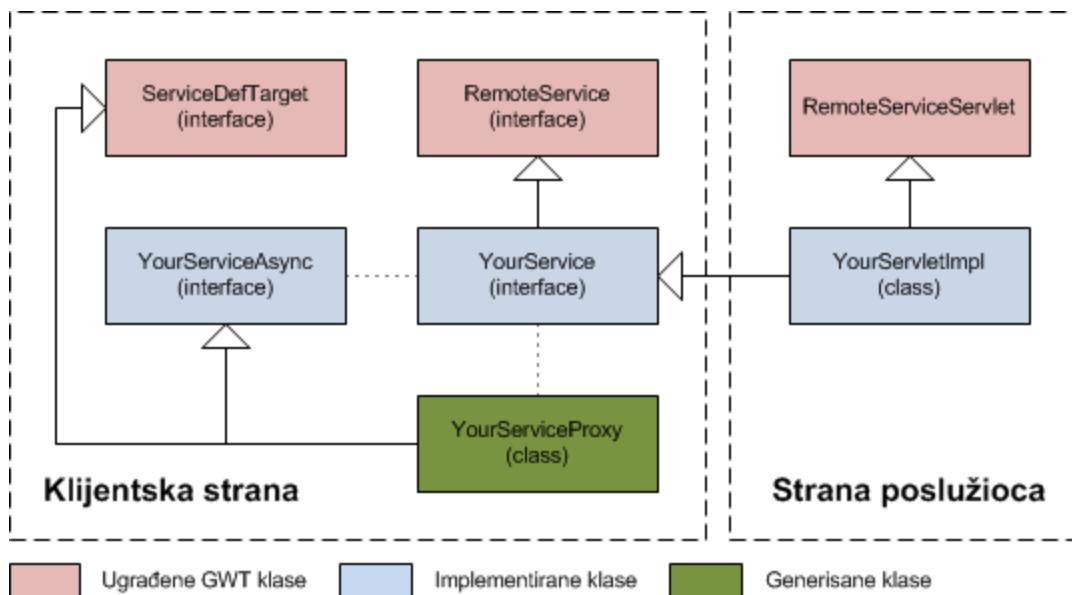
3.2 Deo aplikacije za komunikaciju sa poslužiocem

Većina GWT aplikacija zahteva interakciju sa delom aplikacije koji se izvršava na poslužiocu. GWT nudi nekoliko načina komunikacije sa poslužiocem putem HTTP sprege. Može se koristiti ugrađeni GWT RPC mehanizam koji obezbeđuje pozive ka Java servletima i rukovanje detaljima niskog nivoa kao što je serijalizacija objekata. Osnovna razlika između AJAX i tradicionalnih HTTP web aplikacija je da AJAX aplikacije nemaju potrebe da učitavaju nove HTTP stranice dok se izvršavaju. Međutim, kao i sve klijent/server aplikacije, AJAX aplikacije imaju potrebe za preuzimanjem podataka sa poslužioca dok se izvršavaju. Mehanizam za interakciju sa poslužiocem preko mreže se naziva pozivom udaljene metode (RPC).

Kreiranje RPC se sastoji iz sledećih koraka:

1. Definisanje sprege koja nasleđuje ugrađenu GWT spregu *RemoteService*.
2. Definisanje klase koja implementira kod sa strane poslužioca. Ova klasa treba da nasleđuje ugrađenu GWT klasu *RemoteServiceServlet* i da implementira spregu koja je kreirana pod [1].
3. Definisanje asinhronne sprege koja će se pozivati sa klijentske strane.

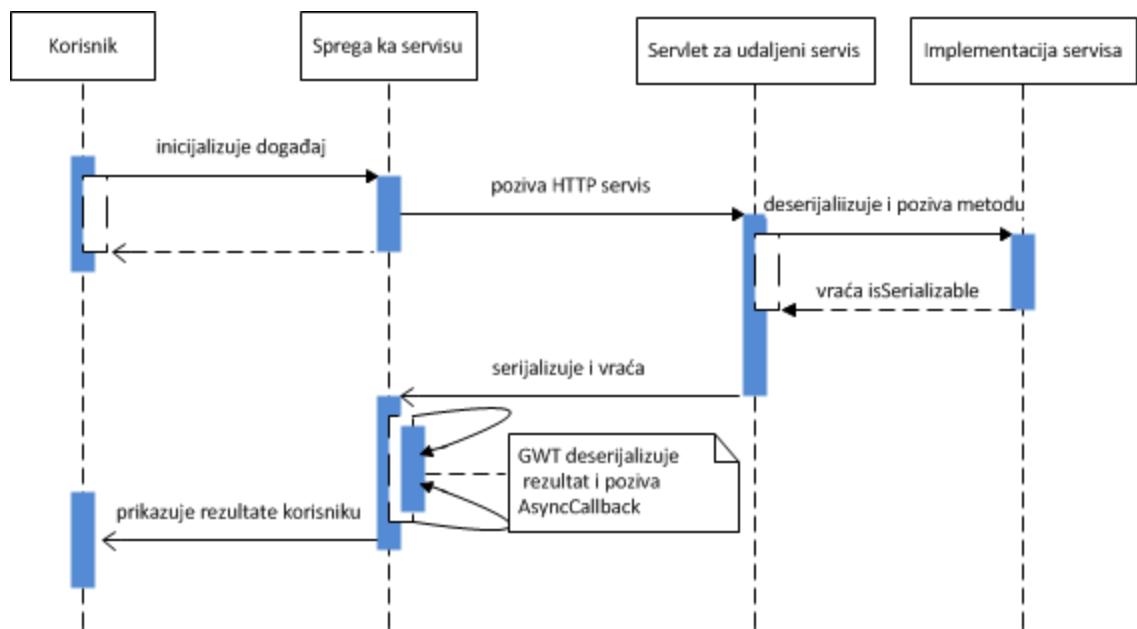
Grafički prikaz realizacije poziva udaljene metode je prikazan na slici 3.4.



Slika 3.4 Dijagram realizacije RPC

Da bi se preko mreže poslali složeni objekti, prethodno treba da se serijalizuju. GWT pruža mehanizam serijalizacije putem ugrađene GWT sprege *Serializable* ili *IsSerializable*. Da bi se neki objekat serijalizovao potrebno je implementirati jednu od sprega.

Redosled komunikacije klijentke aplikacije sa poslužiocem je prikazan na slici 3.5.



Slika 3.5 Dijagram redosleda komunikacije sa poslužiocem

4. Programsко rešenje

U okviru ovog rada svi moduli su urađeni u programskom jeziku Java, korišćene su komponente GWT skupa alata i .xml datoteke. Programsko rešenje je realizovano modularno tako da se svaki modul može vrlo lako izmeniti ili proširiti. Modularnost je obezbeđena korišćenjem MVP projektnog uzorka.

4.1 Moduli i metode GWT aplikacije

GWT aplikacija se sastoji od sledećih paketa:

- com.rtrk.insight.acs.gwt
- com.rtrk.insight.acs.gwt.client – Sadrži početnu tačku aplikacije i skup svih stringova.
- com.rtrk.insight.acs.gwt.client.async – Sadrži sprege ka RPC servletima.
- com.rtrk.insight.acs.gwt.client.engine – Sadrži matične klase i sprege.
- com.rtrk.insight.acs.gwt.client.engine.security – Sadrži klase i sprege koje su neophodne za rukovanje pravima pristupa korisnika.
- com.rtrk.insight.acs.gwt.client.event – Sadrži klase za opis događaja i sprege koje predstavljaju rukovaće događajima.
- com.rtrk.insight.acs.gwt.client.presenter – Sadrži klase koje obezbeđuju logiku svake stranice.
- com.rtrk.insight.acs.gwt.client.view – Sadrži klase i .xml datoteke koje opisuju izgled svake stranice.
- com.rtrk.insight.acs.gwt.servlets.rpc – Sadrži klase koje obezbeđuju komunikaciju sa poslužiocem
- com.rtrk.insight.acs.gwt.shared – Sadrži klase koje su deljene između modula aplikacije.

4.1.1 Paket com.rtrk.insight.acs.gwt

Ovaj paket sadrži modul Main.gwt.xml koji obezbeđuje osnovna podešavanja, kao što su osnovni izgled aplikacije, spisak pretraživača za koje se prevodi izvorni kod, definiciju klase koja predstavlja početnu tačku aplikacije.

4.1.2 Paket com.rtrk.insight.acs.gwt.client

Ovaj paket sadrži module:

- Main – Ova klasa predstavlja početnu tačku aplikacije i sadrži inicijalizacije osnovnih modula bez kojih aplikacija ne bi radila.
- Strings – Predstavlja skup stringova korišćenih u aplikaciji. Namena ovog modula je da se obezbedi jednostavna podrška za lokalizaciju i laku izmenu naziva.

4.1.3 Paket com.rtrk.insight.acs.gwt.client.async

U ovom paketu nalaze se sprege *CpeManagementService* i *CpeManagementServiceAsync*.

Sprega *CpeManagementServiceAsync* poziva se na klijentskoj strani poslužioca.

4.1.4 Paket com.rtrk.insight.acs.gwt.client.engine

Spisak modula ovog paketa je dat u tabeli 4.1.

AppController
AppEventBus
IBasicDisplay
Presenter
StatusManager
View

Tabela 4.1 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.engine

4.1.4.1 AppController

Da bi se upravljalo logikom koja nije određena za Presenter-e i koja pripada aplikativnom sloju uvodi se komponenta *AppController*. Ova komponenta omogućava upravljanje istorijom i promenom stranica pretraživača. Tranzicija pogleda je direktno povezana sa upravljanjem istorijom. U ovoj klasi se definišu sve klase *View* i *Presenter*.

Metode ove klase su:

- public AppController(PermissionHolder holder, ConfigurationServiceAsync rpcInfoService, CpeManagementServiceAsync rpcCpeListService, StatsServiceAsync rpcStats, StatusManager statusManager, AppEventBus eventBus) – Konstruktor.

- `protected void bind()` – U ovoj metodi se pretplaćuje na događaje potrebne za rad aplikacije.
- `private void doLogout()` – Metoda za odjavljivanje korisnika sa poslužioca.
- `public void go(final HasWidgets container)` – Metoda za pokretanje datog modula, ulazni parameter *container* predstavlja prostor u kome će se aplikacija prikazati u pretraživaču.
- `public void onValueChange(final ValueChangeEvent<String> event)` – Pri promeni lokacije pretraživača na osnovu ulaznog parametra *event* proverava se da li korisnik ima pravo da pristupi određenoj stranici. Ako korisnik ima pravo za pristup stranici poziva se metoda *doValueChange*.
- `private void doValueChange(ValueChangeEvent<String> event)` – Na osnovu ulaznog parametra *event* pokreće se odgovarajući Presenter i objavljuje se događaj za promenu lokacije pretraživača, to jest prelazak na drugu stranicu.
- `public void start()` – U ovoj metodi se vrši inicijalizacija svih klasa *Presenter* i *View* i poziva se početna stranica aplikacije.

4.1.4.2 AppEventBus

Namena modula *AppEventBus* je obezbeđivanje jedinstvenog mehanizma komunikacije između Presenter-a. Klasa *AppEventBus* je potomak ugrađene GWT klase *HandlerManager*.

Metode ove klase su:

- `public AppEventBus()` – Konstruktor.
- `public void showError(String message)` – Metoda koja objavljuje događaj za prikaz obaveštenja sa porukom greške.
- `public void showInfo(String message)` – Metoda koja objavljuje događaj za prikaz obaveštenja sa porukom informacije.
- `public void showAlert(String message)` – Metoda koja objavljuje događaj za prikaz obaveštenja sa porukom upozorenja.
- `public void showYesNoDialog(AsyncCallback<Boolean> callback, String message)` – Metoda koja objavljuje događaj za prikaz dijaloga sa porukom.
- `public void showYesNoDialog(AsyncCallback<Boolean> callback, String message, String labelYes, String labelNo)` – Metoda koja objavljuje događaj za prikaz dijaloga sa porukom i sa određenim nazivima dugmadi.
- `public void showLoading()` – Metoda koja objavljuje događaj za prikaz indikatora učitavanja stranice.

- `public void hideLoading()` – Metoda koja objavljuje događaj za sakrivanje prikaza indikatora učitavanja stranice.

4.1.4.3 IbasicDisplay

`IBasicDisplay` predstavlja spregu između komponenata *Presenter* i *View*. Sprega sadrži definicije osnovnih metoda koje svaka sprega *Display* treba da ima.

Metode ove spregi su:

- `void hide()` – Prototip metode predviđen za sakrivanje pogleda.
- `void show()` – Prototip metode predviđen za prikaz pogleda.
- `Widget asWidget()` –
- `List<Composite> getChildren()` – Porototip metode da preuzimanje liste potomaka pogleda.
- `Boolean hasParent()` – Prototip metode za preuzimanje informacije o postojanju potomka.
- `void permissionChange()` – Prototip metode za promenu dozvole.

4.1.4.4 PlaceController

Metode ove klase su:

- `public PlaceController(AppEventBus eventBus)` – Konstruktor.
- `public void goTo(String placeId, @SuppressWarnings("rawtypes") Presenter presenter, HasWidgets container)` – Metoda objavljuje događaj za promenu lokacije pretraživača *PlaceChangedEvent*. Ulazni parametri su *placeId* - jedinstven naziv stranice, *presenter* – određeni presenter stranice i *container* – kontejner u okviru kojeg će se određeni presenter pokrenuti.

4.1.4.5 Presenter

Ovaj modul predstavlja korensku klasu koju nasleđuju svi *Prsenter-i*.

Polja klase su:

- *eventBus* – Magistrala događaja.
- *display* – Sprega za povezivanje modula *Presenter* i *View*.

Metode ove klase su:

- `public Presenter(AppEventBus eventBus, D display)` – Konstruktor.
- `public void go(final HasWidgets container)` – Metoda za pokretanje Presenter-a u okviru kontejnera.
- `public void go()` – Metoda za pokretanje Presenter-a

- `public abstract void start()` – Abstraktna metoda namenjana za pozive RPC metoda.
- `protected abstract void bind()` – Abstraktna metoda namenjena za preplaćivanje na događaje.

4.1.4.6 StatusManager

Ovaj modul omogućuje automatsko ažuriranje podataka o uređajima, programskoj podršci itd. StatusManager sadrži časovnik čiji se period podešava pomoću parametra `REFRESH_INTERVAL`. Pri isteku časovnika upoređuju se lista podataka sa klijentske strane i lista sa strane poslužioca. Ako je došlo do promene nekog parametra, objavljuje se događaj promene statusa uređaja `StatusChangedEvent`.

4.1.4.7 View

Ova klasa sadrži jedno polje `hasParent`, to sadrži informaciju da li ima roditelja, to jest da li se pogled nalazi unutar nekog drugog pogleda.

Metode ove klase su:

- `public View()` – Konstruktor.
- `public Boolean hasParent()` – Metoda za čitanje vrednosti promenljive `hasParent`.
- `public void setHasParent(Boolean hasParent)` – Metoda za postavljanje vrednosti promenljive `hasParent`.

4.1.5 Paket com.rtrk.insight.acs.gwt.client.engine.security

Moduli ovog paketa omogućuju rukovanje pravima prisrupa korisnika.

Spisak sprega:

DevicePermissions
FirmwarePermissions
Permission
PermissionFactory
PermissionHolder

Tabela 4.2 Spisak sprega paketa com.rtrk.insight.acs.gwt.client.engine.security

4.1.6 Paket com.rtrk.insight.acs.gwt.client.event

Klase ovog paketa nasleđuju šablonsku (eng. template) klasu `GwtEvent` sa parametrom određenog rukovaoca događaja (eng event handler). Klasa `GwtEvent` predstavlja koren svih GWT događaja.

Spisak klasa:

Klasa	Opis događaja
CpeListItemSelectedEvent	CPE izabran iz liste CPE.
DeviceListActivatedEvent	Izabrana je stranica sa listom uređaja.
EditUserActivatedEvent	Izabrana je stranica za pregled parametara korisnika poslužioca.
FirmwareApplyEvent	Potvrđeno je ažuriranje programske podrške.
FirmwareSelectedEvent	Programska podrška je izabrana iz liste programskih podrški.
FirmwareUpgradeActivatedEvent	Izabrana je stranica za ažuriranje programske podrške.
LogoutEvent	Korisnik se odjavio sa poslužioca.
ModifyDeviceActivatedEvent	Izabrana je stranica za prikaz i izmenu parametara uređaja.
ModifyDeviceEvent	Promenjen je neki parametar uređaja.
NotificationEvent	Dogadjaj za prikaz poruke ili dijaloga.
PlaceChangedEvent	Promenila se lokacija pretraživača.
RefreshEvent	Stranica se osvežila.
SessionTimeoutEvent	Istekla je sesija.
StatusChangedEvent	Promenilo se stanje uređaja: uključen/isključen u listi uređaja.
StatusMonitorParametersEvent	Istekao je brojač za proveru stanja liste uređaja.
UsersActivatedEvent	Izabrana je stranica za korisnika poslužioca.

Tabela 4.3 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.event

Rukovaoci događaja su sprege u kojima se nalaze prototipovi metoda koji se pozivaju ako se desi određeni događaj. Metode implementira klasa koja je pretplaćena na određeni događaj.

Spisak sprega:

Sprega
CpeListItemSelectedEventHandler
DeviceListActivatedEventHandler
EditUserActivatedEventHandler

FirmwareApplyEventHandler
FirmwareSelectedEventHandler
FirmwareUpgradeActivatedEventHandler
LogoutEventHandler
ModifyDeviceActivatedEventHandler
ModifyDeviceEventHandler
NotificationEventHandler
PermissionsReceivedEventHandler
PlaceChangedEventHandler
RefreshEventHandler
SessionTimeoutEventHandler
StatusChangedEventHandler
StatusDeviceChangedEventHandler
StatusEventHandler
StatusGraphChangeEventHandler
StatusMonitorParametersEventHandler
UsersActivatedEventHandler

Tabela 4.4 spisak sprega paketa com.rtrk.insight.acs.gwt.client.event

4.1.7 Paket com.rtrk.insight.acs.gwt.client.presenter

U ovom paketu nalaze se potomci šablonске klase *Presenter*. Svaka klasa kao polje ima spregu *Display* koja služi za povezivanje modula *Presenter* sa odgovarajućim modulom *View*.

Klasa	Opis
CpeInfoPresenter	Predstavlja logiku za prikaz osnovnih informacija izabranog uređaja
CpeListPresenter	Predstavlja logiku za prikaz liste uređaja.
DeviceListPresenter	Predstavlja logiku za prikaz stranice sa listom uređaja.
EditCpePresenter	Predstavlja logiku za izmenu parametara uređaja.
EditUserPresenter	Predstavlja logiku za izmenu parametara korisnika.
FirmwareListPresenter	Predstavlja logiku za prikaz liste dostupnih programskih podrški za ažuriranje na odabranoj grupi CPE
FirmwareUpgradePresenter	Predstavlja logiku za prikaz stranice za ažuriranje programske podrške.
MainPresenter	Sadrži logiku za prikaz osnovnog izgleda stranice.

MenuPresenter	Sadrži logiku za prikaz menija.
ModifyDevicePresenter	Sadrži logiku za prikaz stranice za postavljanje parametara izabranog CPE.
NotificationPresenter	Sadrži logiku za prikaz obaveštenja.
OwnerInfoPresenter	Sadrži logiku za prikaz podataka o vlasniku izabranog CPE.
TopPresenter	Sadrži logiku za prikaz trenutno prijavljenog korisnika na poslužiocu.
UsersPresenter	Sadrži logiku za prikaz stranice sa listom korisnika.

Tabela 4.5 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.presenter

4.1.8 Paket com.rtrk.insight.acs.gwt.client.view

U ovom paketu definisani su izgledi stranica grafičke korisničke sprege. Za svaku stranicu postoji jedna klasa u Java programskom jeziku koja definiše osnovne komponente kao što su labele, dugmad itd. i odgovarajuća .xml datoteka koja definiše izgled, raspored i poziciju komponenata. Klase ovog paketa su potomci klase *View*, povezuju se sa odgovarajućim *Presenter*-om preko sprege *Display*. U tabeli 4.6 nalazi se spisak klasa ovog paketa.

Klasa
CpeInfoView
CpeListView
DeviceListView
EditCpeView
FirmwareListView
FirmwareUpgradeView
MainView
MenuView
ModifyDeviceView
NotificationView
OwnerInfoView
TopView
UsersView

Tabela 4.6 Spisak klasa paketa com.rtrk.insight.acs.gwt.client.view

4.1.8.1 CpeInfoView

CpeInfoView modul obezbeđuje prikaz osnovnih podataka o izabranom uređaju iz tabele uređaja. Prikazuju se status uređaja, serijski broj, ime, proizvođač i produktna klasa.

Brief Information



Serial Number: 456456
Name: device_id_4
Manufacturer: RT-RK
OUI:
Product Class: SABC4RS

Slika 4.1 CpeInfoView

4.1.8.2 CpeListView

CpeListView modul obezbeđuje prikaz liste uređaja, pretraživanje uređaja po nekom kriterijumu, brisanje uređaja iz liste. U listi su prikazani najosnovniji parametri kao što su status uređaja uključen/isključen, proizvođač i datum i vreme poslednje aktivnosti uređaja. Lista je realizovana tako da bude ponovo upotrebljiva, podržano je biranje više uređaja u isto vreme. To se postiže postavljanjem promenljive *multiselection* na *true*. Da se poslužioc ne bi opteretio ne prikazuju se svi uređaji u listi odjednom nego se podaci prikazuju u grupama od 10 ili 30 uređaja. Na dnu stranice nalaze se navigacioni dugmići koji obezbeđuju prikaz prve, prethodne, sledeće ili poslednje grupe uređaja.

Device List		► Refine By	:	<input type="text"/>	10
	Name	Manufacturer	Last Seen		
	Beograd	RT-RK	2013-06-12 10:52		
	device_id_1	RT-RK	2013-06-12 19:05		
	device_id_10	RT-RK	2013-06-11 18:30		
	device_id_11	RT-RK	2013-06-11 18:30		
	device_id_12	RT-RK	2013-06-11 18:30		
	device_id_13	RT-RK	2013-06-11 18:30		
	device_id_14	RT-RK	2013-06-11 18:30		
	device_id_15	RT-RK	2013-06-11 18:30		
	device_id_3	RT-RK	2013-06-11 14:01		
	device_id_4	RT-RK	2013-06-12 20:38		

Slika 4.2 CpeListView

4.1.8.3 DeviceListView

Ovaj modul obezbeđuje kontejner za prikaz liste uređaja *CpeListView*, prikaz osnovnih podataka o uređaju *CpeInfoView*, i osnovnih podataka o korisniku uređaja *OwnerInfoView*. U njemu se definiše raspored datih pogleda.

Device List				► Refine By	:	<input type="text"/>	10
	Name	Manufacturer	Last Seen				
	device_id_0	RT-RK	2013-06-18 11:05				
	device_id_1	RT-RK	2013-06-18 21:46				
	device_id_2	RT-RK	2013-06-18 21:16				
	device_id_3	RT-RK	2013-06-18 14:04				

Brief Information

Serial Number: -
Name: -
Manufacturer: -
OUI: -
Product Class: -

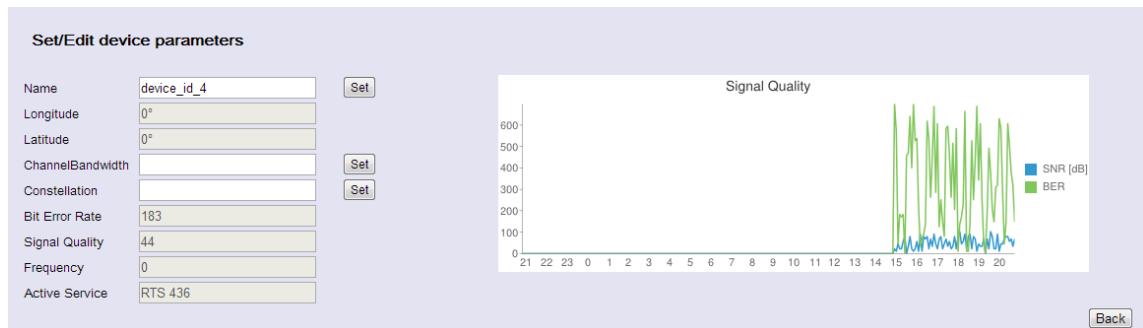
ID :
Name :

Provider :
Address :
Phone :
email :

Slika 4.3 DeviceListView

4.1.8.4 EditCpeView

Ovaj modul sadrži sve parametre koji se mogu pročitati sa poslužioca. U njemu je mogućen pregled i postavljanje vrednosti parametara. Pritiskom na dugme *set* izmenjuje se parametar, a pritiskom na dugme *back* pretraživač se vraća na listu uređaja. U modulu je sa desne strane prikazan grafikon koji prikazuje kvalitet signala i BER.



Slika 4.4 EditCpeView

4.1.8.5 FirmwareListView

Ovaj modul obezbeđuje prikaz liste dostupnih programskih paketa, brisanje programskog paketa iz liste i dodavanje novog programskog paketa u listu. Pritiskom na dugme *Choose File* pojavljuje se novi prozor gde korisnik pronalazi datoteku sa programskim paketom, a pritiskom na dugme *Upload* programski paket se stavlja na poslužioca i u listu dostupnih programskih paketa.

The screenshot shows the 'FirmwareListView' interface. At the top, there is a header bar with the title 'Firmware Upgrade' on the left, a 'Choose File' button with the message 'No file chosen' in the middle, and an 'Upload' button on the right. Below the header is a table listing three firmware packages:

	Firmware	Type	Size	Upload Time
<input type="checkbox"/>	sabc4rsUpgrade_v2.tar	application/octet-stream	7.54MB	2013 Jun 6
<input type="checkbox"/>	sabc4rsUpgrade_v3.tar	application/octet-stream	7.54MB	2013 Jun 6
<input type="checkbox"/>	sabc4rsUpgrade_vTest.tar	application/octet-stream	7.54MB	2013 Jun 10

Slika 4.5 FirmwareListView

4.1.8.6 FirmwareUpgradeView

Ovaj modul obezbeđuje kontejner za prikaz liste dostupnih programskih paketa *FirmwareListView* i listu aktivnih uređaja *CpeListView*. U donjem desnom uglu nalaze se dva dugmeta: *cancel* za otkazivanje ažuriranja programskog paketa i *Apply* za potvrdu ažuriranja programskog paketa.

Firmware Upgrade

	Firmware	Type	Size	Upload Time
<input type="checkbox"/>	sabc4rsUpgrade_v2.tar	application/octet-stream	7.54MB	2013 Jun 6
<input type="checkbox"/>	sabc4rsUpgrade_v3.tar	application/octet-stream	7.54MB	2013 Jun 6
<input type="checkbox"/>	sabc4rsUpgrade_vTest.tar	application/octet-stream	7.54MB	2013 Jun 10

Choose File: No file chosen Upload

Refine By: 10

Device List

	device_id_4	STLinux2.4	2013-06-12 20:52
	Flash Debug Veljko	STLinux2.4	Wed May 29 17:33:00 CEST 2013
	device_id_4	STLinux2.4	2013-06-12 20:52

Cancel Apply

Slika 4.6 Firmware Upgrade View

4.1.8.7 MainView

Ovaj modul sadrži izgled aplikacije koji je vidljiv u pretraživaču. Sastoji se iz tri dela: *TopView*, *MenuView* i u sredini nalazi se komponenta *VerticalPanel* koji predstavlja telo. U telo se dodaje sadržaj određene stranice na primer *DeviceListView*, *ModifyDeviceView* itd.

INSIGHT
Diagnostic and Analytics

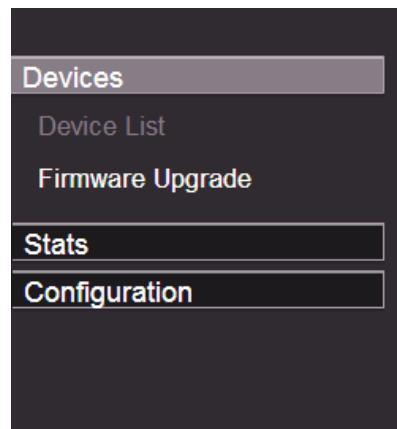
admin : log out

- Devices**
 - Device List
 - Firmware Upgrade
- Stats**
- Configuration

Slika 4.7 MainView

4.1.8.8 MenuView

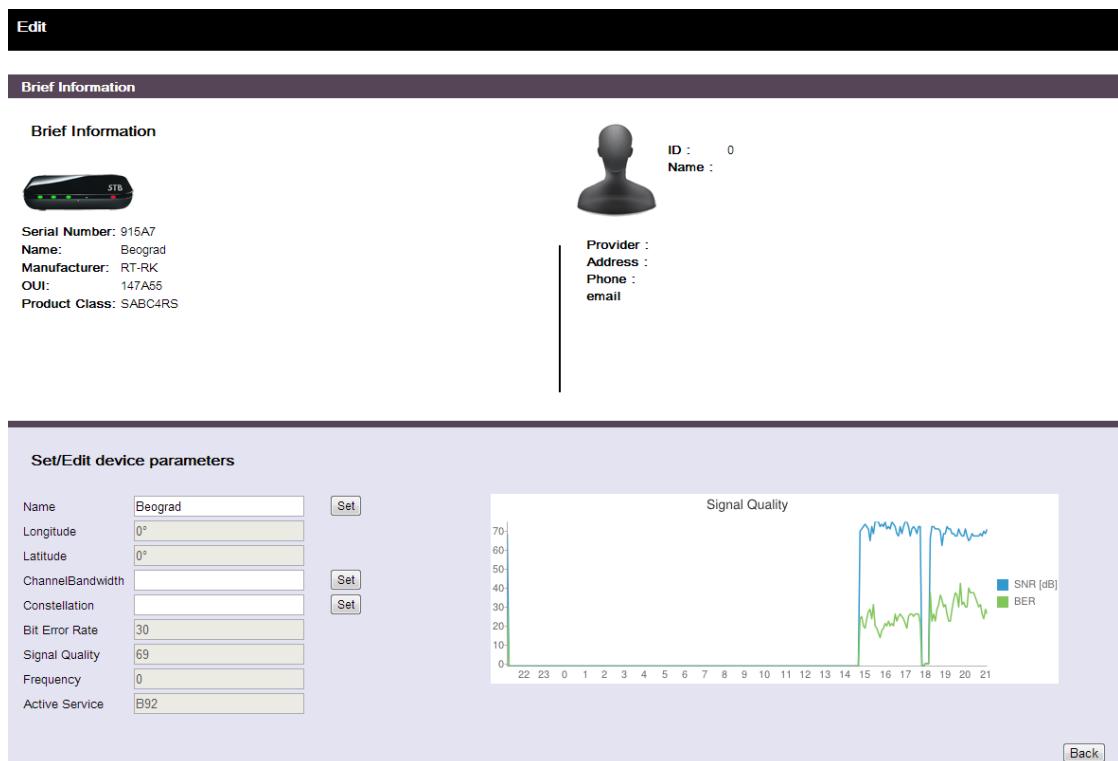
Za realizaciju menija je korišćena komponenta *StackLayoutPanel*. Postoje tri glavne grupe koje su predstavljene kao zaglavlje. Odabirom zaglavlja otvara se padajući meni sa dostupnim stranicama datog zaglavlja.



Slika 4.8 MenuView

4.1.8.9 ModifyDeviceView

Ovaj modul obezbeđuje kontejner za prikaz osnovnih podataka o uređaju *CpeInfoView*, osnovnih podataka o korisniku uređaja *OwnerInfoView* i prikaz i mogućnost postavljanja parametara *EditCpeView*. U njemu se definiše raspored datih pogleda.



Slika 4.9 ModifyDeviceView

4.1.8.10 NotificationView

Ovaj modul obezbeđuje prikaz poruka i dijaloga.

4.1.8.11 OwnerInfoView

Ovaj modul sadrži prikaz osnovnih podataka o vlasniku uređaja kao što su ime, prezime, adresa itd.



Slika 4.10 OwnerInfoView

4.1.8.12 TopView

Ovaj modul predstavlja zaglavlje aplikacije. Prikazan je logo i u gornjem desnom uglu nalazi se dugme *log out*. Pritiskom na dugme *log out* korisnik se odjavljuje sa poslužioca.



Slika 4.11 TopView

4.1.8.13 UsersView

Ovaj modul predstavlja prikaz liste registrovanih korisnika poslužioca i mogućnost izmene parametara korisnika.

	User first name	User e-mail	Access level	User active?
	Paul	admin@rt-rk.com	admin	true

Edit

First name:

Last name:

User e-mail:

Phone number:

Language:

User active?:

User credentials

Username:

Password:

Confirm password:

Access level:

Apply **Cancel**

Slika 4.12 Users View

4.1.9 Paket com.rtrk.insight.acs.gwt.servlets.rpc

U ovom paketu nalazi se klasa *ConfigurationServiceImpl* koja implementira spregu *CpeManagementService* iz paketa com.rtrk.insight.acs.gwt.client.async. Ovaj modul je zadužen za komunikaciju sa poslužiocem.

4.1.10 Paket com.rtrk.insight.acs.gwt.shared

U ovom paketu nalaze se klase koje su deljene između modula sistema. Nalaze se konstante, definicije parametara, klase za čuvanje podataka koje se mogu serijalizovati radi čuvanja u bazi podataka, klase za razmenu podataka između klijenta i poslužioca. Spisak klasa je dat u tabeli 4.7.

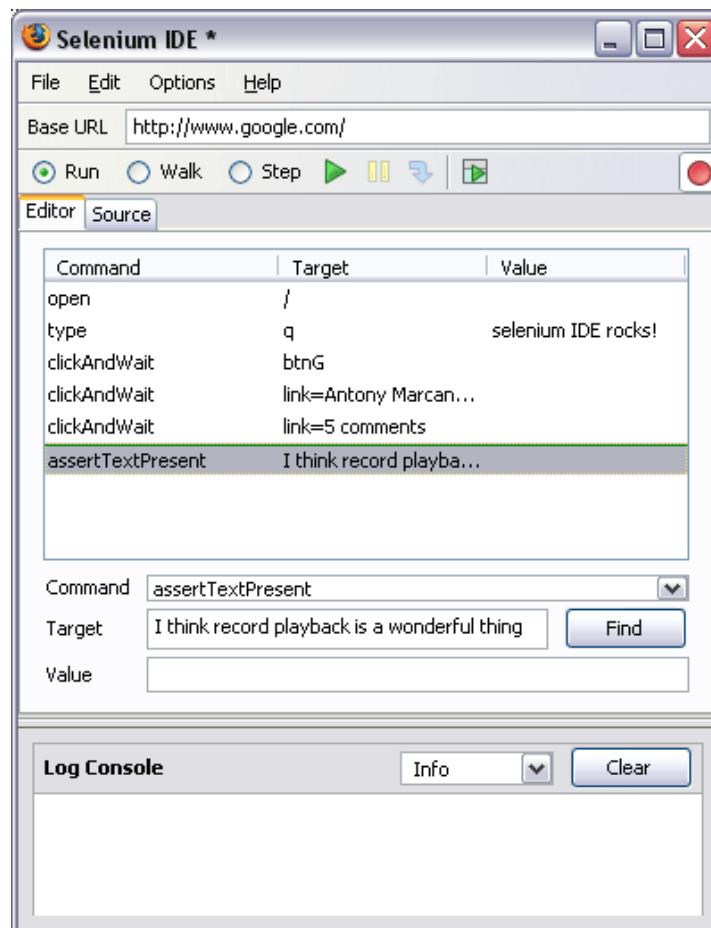
Constants
Cpe
CPEListRequest
CPEListResponse
CPEListStruct
CpeParametarsShared
CPEParameters
CPEParametersRequest
CPEParametersResponse
CPESetParamRequest
CPESetParamResponse
CpeShared

CPEStatusRequest
CPEStatusResponse
Defines
DeviceIdStructRequest
DeviceIdStructShared
Firmware
LoginRequest
LoginResponse
OwnerInfoShared
ParameterShared
QueryShared
RequestObject
responseObject
Role
RoleShared
SearchCPEListRequest
SearchCPEListResponse
SignalParametersShared
User
UserDetails
UserDetailsShared
UserShared

Tabela 4.7 Spisak klasa paketa com.rtrk.insight.acs.gwt.shared

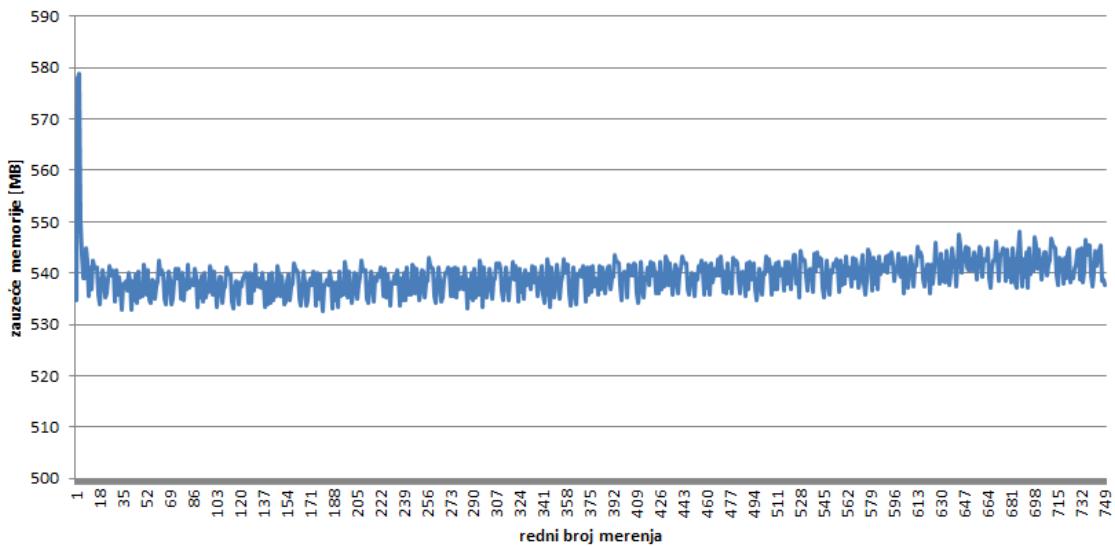
5. Rezultati

Korisnička aplikacija je ispitivana u realnim uslovima kao elementarni deo konfiguracionog poslužioca, i dodatno pomoću specijalizovanih alata za ispitivanje aplikacija baziranih na web tehnologijama. Za ispitivanje je korišćeno razvojno okruženje Selenium IDE slika 5.1, koje predstavlja dodatak pretraživača Firefox i omogućava automatizaciju pretraživača za svrhe ispitivanja i razvojni alat Firebug, koji takođe predstavlja dodatak pretraživača Firefox.



Slika 5.1 Selenium IDE

Pomoću alata Selenium IDE ispitivana je stabilnost i potrošnja memorije aplikacije. Ispitni slučaj sastoji se iz periodičnog menjanja stranice za prikaz liste uređaja *cpeList*, stranice za podatke o CPE *modifyDevice* i stranice za ažuriranje programske podrške *fwUpgrade*. Ispitni slučaj sastoji se takođe iz kretanja kroz liste i postavljanja parametara CPE. Tokom ispitivanja beležilo se zauzeće memorije pretraživača, izvršeno je 750 merenja, prosečno zauzeće memorije je 539.480349 MB. Na slici 5.2 se može videti da postoji curenje memorije koje predstavlja manji problem. Curenje memorije je uočeno pri korišćenju asinhronih zahteva, što je čest problem kod realizacije na današnjim web pretraživačima.



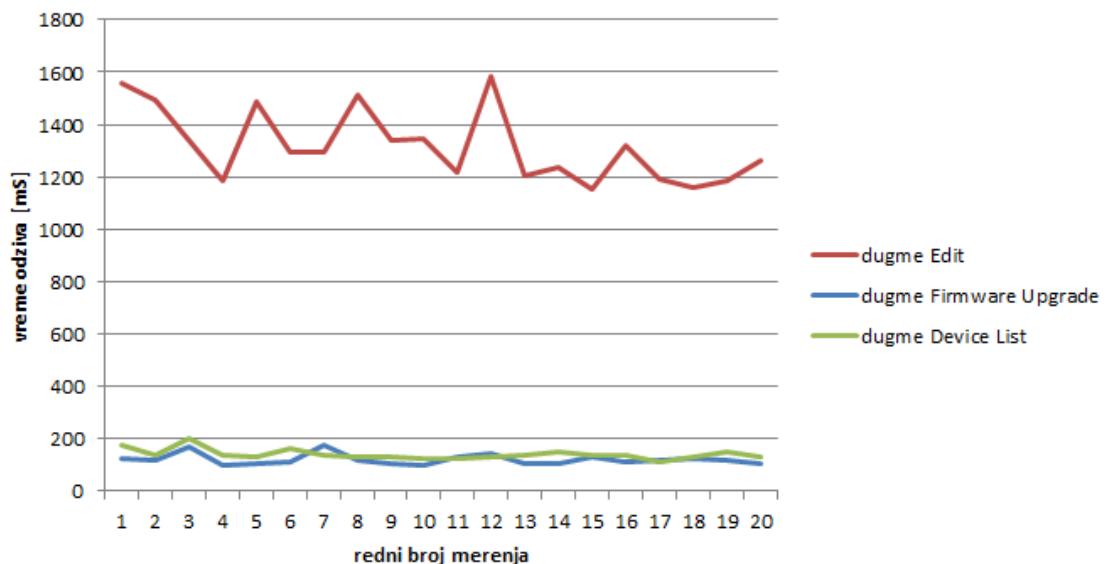
Slika 5.2 Rezultati ispitivanja

Pomoću alata Firebug ispitivano je vreme odziva na komande iz aplikacije do prikaza rezultata na ekranu.

Ispitivanje vremena odziva na komande iz aplikacije je vršeno za tri slučaja:

1. Korisnik sa stranice *cpeList* izabere jednog uređaja iz liste i pritiskom na dugme *Edit* prelazi na stranicu *modifyDevice* za detaljan prikaz podataka o uređaju.
2. Korisnik pritiskom dugmeta iz menija *Firmware Upgrade* otvara stranicu za ažuriranje programskog paketa *fwUpgrade*.
3. Korisnik pritiskom dugmeta iz menija *Device List* otvara stranicu za prikaz liste uređaja *cpeList*.

Sa slike 5.3 se može zaključiti da je zadovoljeno pravilo 2 sekunde [6].



Slika 5.3 Vreme odziva na komande iz aplikacije

Funkcionalno ispitivanje je vršeno u realnim uslovima uz pomoć Insight ACS 1.0 poslužioca i set-top-box uređaja sa podrškom za TR-069 protokol. Ispitivan je prikaz liste uređaja i ažuriranje programske podrške.

Na osnovu dobijenih rezultata potvrđena je ispravnost rada i robustnost aplikacije.

6. Zaključak

Zadatak ovog rada je bio da se realizuje klijentska aplikacija za pristup poslužiocu za konfigurisanje STB korišćenjem MVP projektnog uzorka. Programsko rešenje je realizovano modularno da bi se aplikacija lako proširivala i menjala. Kako je jedna od osobina ovog sloja laka proširivost, budući rad na ovom rešenju će obuhvatiti i kreiranje stranica sa prikazom mape pokrivenosti signalom i stranice za statistiku. Uz pomoć statistike aplikacija će imati analitički značaj.

Rešenje je ispitano korišćenjem Insight ACS 1.0 i set-top boks uređaja sa podrškom za TR-069 protokol.

Optimizacija po pitanju brzine i trošenja memorije je moguća i biće takođe predmet daljeg istraživanja i rada.

7. Literatura

- [1] DSLHome-Technical Working Group: *CPE WAN Management Protocol*, May 2004
- [2] Marketing Report *TR-069 Deployment Scenarios*, Issue: 1, August 2010
- [3] Sajt <https://developers.google.com/web-toolkit/>
- [4] tutorialspoint.com: *GOOGLE WEB TOOLKIT TUTORIAL*
- [5] Google Web Toolkit Solutions: *Cool & Useful Stuff*, David Geary, 2007
- [6] Response time in man-computer conversational transactions, Miller, R. B., 1968